



Rapport de Stage KOPADIA

Commande de drones sous-marins FOLAGA

PÉRIODE : 14 mai 2018 – 3 août 2018

MAÎTRE DE STAGE : Hervé RICHER DE FORGES

TUTEUR ÉCOLE : Luc JAULIN

STAGIAIRE : Paul-Antoine GRAU

PROFIL ROBOTIQUE



Résumé

Ce rapport présente le stage que j'ai effectué pendant 12 semaines dans la start-up KOPADIA, une entreprise qui cherche à se positionner sur le marché de l'inspection sous-marine et qui est basée à Orsay à côté du plateau de Saclay.

Ce stage a pour but de réaliser un suivi de fond à l'aide de drone sous-marins autonomes. A l'origine, la commande des drones ne se faisait qu'avec des outils du constructeur. Il fallait développer toute une architecture qui lit les données nécessaires, qui les traite et envoie une commande au drone.

Je commence donc par l'intégration d'un capteur qui mesure la distance entre le drone et le fond marin. Cette intégration est aussi bien matérielle que logicielle. Après avoir accédé à cette donnée, il fallait la partager avec d'autres applications, dans le but d'obtenir une commande, à l'aide d'un middleware – MOOS.

A la suite de ce développement via ce middleware, il a fallu corriger l'assiette du drone pour que celui-ci reste le plus à plat possible et obtenir des meilleurs résultats. Une fois cela fait, nous avons réglé le drone en altitude comme souhaité par l'entreprise.

Abstract

This report shows the internship I did for 12 weeks in the start-up named KOPADIA, a company which wants to capture the underwater inspection market. It is based in Orsay, near the plateau of Saclay.

This internship aim to develop a seabed tracking with autonomous underwater vehicles. At the beginning, the command of the drones were only with tools of the builder. I had to design an architecture which read the wanted data, which process its and send them to the drone.

So, I begin with the integration of a sensor which measure the distance between the drone and the seabed. This integration has to be hardware and software as well. After reading the data, I had to share the data with others applications – to have a command after processing – with the middleware MOOS.

Therefore the development with the middleware I had to track the pitch of the drone because we wanted the drone to work on an horizontal plane to have better results. Once it was done, we had to set the seabed tracking as wished by the start-up.

Remerciements

Je tiens à remercier mon professeur de robotique et tuteur de stage, Monsieur Luc JAULIN pour m'avoir permis de trouver ce stage qui fut très intéressant et qui correspondait parfaitement à ce que je cherchais. Merci !

Je souhaite aussi exprimer tous mes remerciements à mon maître de stage Monsieur Hervé RICHER DE FORGES pour m'avoir fait confiance du premier au dernier jour de mon stage pour tout le développement des algorithmes de commande. Il m'a permis de m'exprimer pleinement et m'a aidé ainsi que toute l'équipe de KOPADIA à me mettre en confiance pour pouvoir développer efficacement des algorithmes.

Je tenais également à exprimer ma reconnaissance à toute l'équipe qui m'a intégré très rapidement dans cette entreprise. Thierry GROUSSET, Hervé RICHER DE FORGES, Agnès LACROIX, Gaëtan FONTAINE, Romain DUVAL, Oriane ADAMM, ils ont tous été très sympathiques et ouverts d'esprit. Cela a aidé et a permis de prendre des décisions, faire des choix comme l'architecture de développement, entre autres, de manière plus efficace.

Table des matières

1	Le contexte	5
2	Les objectifs et les enjeux	7
3	Activités et résultats	8
3.1	Organisation	8
3.2	Présentation des FOLAGA	8
3.2.1	Les caractéristiques techniques	8
3.2.2	Les déplacements	8
3.2.3	Le contrôle du drone	9
3.2.4	Les <i>payload</i>	10
3.3	L'écho sondeur	11
3.3.1	Son fonctionnement	11
3.3.2	L'intégration matérielle	13
3.3.3	L'intégration logicielle	13
3.3.4	Résultats	13
3.4	Mission Oriented Operating Suite - MOOS	14
3.4.1	Le fonctionnement	15
3.4.2	Architecture choisie	16
3.5	Régulation d'assiette	18
3.5.1	Motivations et cahier des charges	18
3.5.2	Réglage du PID	18
3.6	Suivi de fond	19
3.6.1	Une première simulation	19
3.6.2	Une deuxième simulation	20
3.6.3	En réel	22
4	Analyse économique	24
5	Projet professionnel	25
6	Conclusion	26

Table des figures

1	FOLAGA équipé de capteurs	6
2	FOLAGA dans le lac Polytechnique	6
3	Diagramme de GANNT	8
4	Les pompes de déplacement	9
5	La GUI permettant le contrôle des drones	9
6	Payload acoustique	10
7	Écho sondeur Imagenex 852 Ultra-miniature	11
8	Exemple de trame émise à l'écho sondeur	11
9	Altitude du drone lue à Brest lors d'une mission	14
10	Zoom sur l'altitude du drone lue à Brest	14
11	Fonctionnement de MOOS	15
12	Exemple de fichier mission du test de la lecture sondeur	16
13	Architecture de développement de MOOS	17
14	Graphique de la régulation en assiette	19
15	Simulation du suivi de fond par l'AUV	20
16	Suivi de fond simulé avec une dynamique corrigée	20
17	Bilan des forces appliquées au drone	21
18	Comparaison de l'altitude lue par le suivi de fond avec et sans filtre	22
19	Suivi de fond régulé par un PID – Mission de 90 secondes	23
20	Suivi de fond régulé par un PID – Mission de 180 secondes	23

Liste des tableaux

1	Influence de l'assiette sur l'altitude du drone	18
2	Coefficients finaux de réglage du PID de la régulation d'assiette	19
3	Masse lue sur le pèse main en fonction de la propulsion verticale	22
4	Coefficients finaux de réglage du PID du suivi de fond	23

1 Le contexte

La robotique s'est amplement développée et démocratisée ces dernières années, notamment dans les domaines aérien – avec les drones – et terrestre – avec les véhicules autonomes. Le domaine sous-marin est un peu plus en reste que les deux précédents même s'il est actuellement en pleine expansion.

Il existe aujourd'hui une communauté scientifique qui le façonne et l'utilise : environ 4 000 bouées nommées ARGO[2] utilisent le système de positionnement ARGOS¹ pour effectuer des mesures océanographiques et climatologiques. De plus, une flotte de 300 *gliders*² explore aujourd'hui les fonds maritimes. Toujours dans l'exploration, en janvier 2018, l'entreprise OCEAN INFINITY a recommencé à chercher le MH370³[3] à l'aide d'une flotte de 8 drones autonomes sous-marins – aussi appelés AUV⁴ – des *Hugin 6000* qui coûtent environ 10 millions d'euros pièce.

Il existe beaucoup de projets pour l'avenir et de systèmes déjà présents pour des applications dans l'industrie pétrolière. Par exemple, l'entreprise Brésilienne PETRUS a récemment – en Février 2018 – signé un partenariat avec ECAGROUP[5] pour poser, inspecter et entretenir les pipelines et câbles sous-marins. Pour cela, PETRUS utilisera également des drones sous-marins.

Ce marché d'inspection et d'exploration pour la pose et l'entretien des pipelines est en plein essor en raison de l'augmentation des incidents sur les pipelines puisque celles-ci arrivent en fin de vie. Il faut savoir qu'il y a 160 000 km de pipelines sous-marines – ce qui représente 4 fois le tour de la Terre. Pour une bonne partie d'entre elles, la durée de vie théorique est dépassée. C'est pourquoi l'ensemble du milieu sous-marin a besoin de la robotique.

C'est dans ce contexte que KOPADIA, une start-up âgée seulement d'un an et demi, développe des solutions d'inspection sous-marine et d'ingénierie. Elle cherche à se positionner sur le marché pétrolier mais aussi sur celui de l'éolien maritime – par l'inspection des câbles électriques – et par la même occasion des télécommunications – inspection des câbles réseaux. Elle souhaite aussi se développer pour avoir la capacité de mesurer les impacts environnementaux.

KOPADIA est située à côté du plateau de Saclay dans le Bassin Parisien, à Orsay, dans l'incubateur INCUBALLIANCE. Sept personnes travaillent dans cette entreprise :

- un président co-fondateur, Thierry GROUSSET
- un directeur technique co-fondateur, Hervé RICHER DE FORGES
- une office manageuse⁵, Agnès LACROIX
- un ingénieur en conception de systèmes mécaniques, Gaëtan FONTAINE
- un élève de l'ISEN Brest, Romain DUVAL en contrat de professionnalisation en robotique
- une stagiaire de l'ISEN Brest en 4ème année en systèmes embarqués, Oriane ADDAM

¹Système permettant un positionnement précis à environ 150 m

²Les gliders sont des planeurs sous-marins

³Avion tristement connu car il est porté disparu depuis 2014

⁴Autonomous Underwater Vehicle

⁵Une office manager use gère toutes les tâches administratives, juridiques et comptables

Pour inspecter les océans, KOPADIA possède des drones sous-marins autonomes - qui ont été nommés FOLAGA par leur constructeur italien GRAAL TECH - sur lesquels elle commence à ajouter des capteurs qui permettent et permettront d'étudier précisément les fonds marins. Parmi ces capteurs, on trouve, une tête et un système acoustiques qui permettent au drone de se localiser sous l'eau, un sideScan⁶ et une caméra.



FIGURE 1 : FOLAGA équipé d'un sideScan, de la tête acoustique et d'une caméra

Pour tester les différentes missions que l'entreprise développe, un petit bassin d'essai - environ 4m de long pour 2m de large et 1m20 de profondeur - est installé à proximité de l'atelier, l'endroit où sont développés les algorithmes de navigation. Pour de plus amples essais, un partenariat avec l'école de Polytechnique permet d'avoir accès à leur lac. Celui-ci est suffisamment étendu pour tester à échelle réelle les solutions actuellement développées. Cependant, comme il ne fait que 2 m de profondeur au maximum, il ne permet pas de vérifier correctement les solutions qui nécessitent plus de fond. Pour tester dans des eaux plus profondes, KOPADIA se déplace parfois à la *Sea Test Base* en rade de Brest.



FIGURE 2 : FOLAGA dans le lac Polytechnique

De plus, dans le cadre de la recherche de la Cordelière – sujet proposé par Monsieur Luc JAULIN – KOPADIA est amenée à se déplacer à Brest pour tester les solutions et tenter de retrouver le bateau naufragé. Lors de mon stage nous nous sommes déplacés une fois à Brest pendant 2 jours.

⁶Le sideScan mesure les profondeurs du fond sous-marin sur une portion de sol

2 Les objectifs et les enjeux

KOPADIA a donc pour but de vendre des solutions d'inspection sous-marine. Il lui est nécessaire pour son développement d'implémenter des algorithmes de navigation autonome. Étant une très jeune entreprise avec un seul roboticien en contrat de professionnalisation – Romain DUVAL – il n'existe pas d'algorithme de navigation propre à KOPADIA lors de mon arrivée.

Très vite, je suis orienté vers la conception et le développement d'un suivi de fond. Dans un premier temps, je dois intégrer un sondeur pour connaître l'altitude du drone et ainsi pouvoir le réguler par rapport au fond. C'est un algorithme clef car il est nécessaire au bon fonctionnement de tous les capteurs d'inspection : au-delà d'une certaine altitude, ils peuvent ne pas fonctionner correctement. Par exemple, le sideScan et la caméra sont des capteurs qui ne renverront pas des données exploitables si la distance entre le fond marin et l'AUV est trop importante ou trop faible.

Ma mission est donc importante pour le développement – dans un premier temps – technologique, mais aussi bien économique dans un second temps, de l'entreprise. C'est pourquoi Romain et Oriane sont eux aussi mobilisés, pour développer l'ensemble des briques technologiques permettant de récupérer des données du FOLAGA, de les traiter, et de contrôler le drone.

Pour KOPADIA, l'intégration de l'écho Sondeur est la première intégration d'un capteur propre à l'entreprise et aussi du premier développement d'algorithme de navigation. Cette société est donc peu expérimentée dans l'intégration des capteurs et dans la réalisation d'une commande. Cependant, les connaissances de chacun rendent cette intégration possible. C'est un challenge.

D'un point de vu personnel, c'est la première fois que je dois intégrer un capteur à un système, et nous n'avons que 3 jours pour l'associer au drone. C'est donc également un challenge pour moi. Ayant participé au projet d'UV 2.4 Positionnement GPS lors de ma formation, j'ai quelques connaissances quant à la lecture d'un capteur, des ports séries et des trames. Voilà pourquoi l'intégration du sondeur est confiée à Romain et à moi-même – Oriane n'ayant pas encore commencé son stage.

3 Activités et résultats

Dans cette partie, je vais dans un premier temps présenter les différents supports sur lesquels j'ai travaillé pendant ce stage pour permettre une vision globale du système ce qui facilitera l'explication de ce que j'ai apporté techniquement à l'entreprise pendant ces 12 semaines de stage. Tout d'abord, voici un diagramme de GANTT présentant la répartition des tâches que j'ai effectuées pendant mon stage.

3.1 Organisation

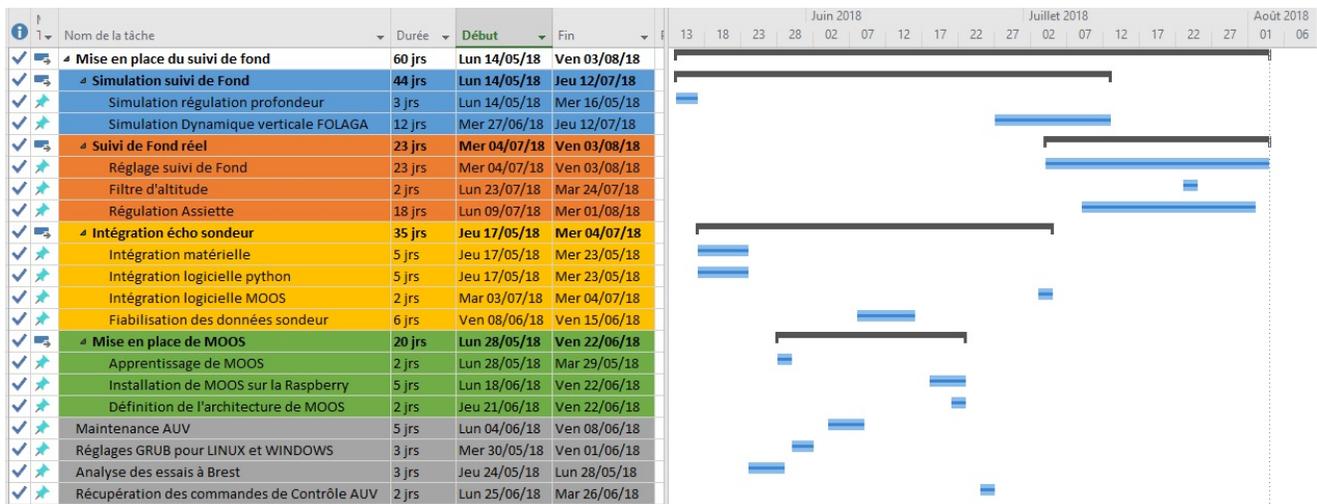


FIGURE 3 : Diagramme de GANTT

3.2 Présentation des FOLAGA

3.2.1 Les caractéristiques techniques

Les drones mesurent entre 2 et 3m. Leur taille peut varier grâce à des sections optionnelles appelées *payload*. Ils pèsent environ 30 kg dans l'air. A l'aide d'une chambre de pressurisation, ils sont neutres dans l'eau – ils pèsent donc 0 kg dans l'eau. Ils peuvent avancer avec une vitesse maximale d'environ 2 nœuds – soit 1 m/s – dans le sens de poussée et environ 0,2 m/s en vertical. Théoriquement, ils possèdent une autonomie de 6 heures et peuvent descendre jusqu'à 60 m de profondeur. De plus ils sont équipés d'une antenne WiFi et GPS.

3.2.2 Les déplacements

Les drones FOLAGA sont équipés de 4 pompes à l'avant et de 4 autres à l'arrière. Pour l'avant ainsi que pour l'arrière, 2 de ces pompes permettent aux drones de se déplacer dans un plan horizontal tandis que les 2 autres leur offrent la possibilité de se déplacer selon un plan vertical (voir figure 4). De plus ils possèdent une propulsion avec une hélice qui leur permet d'avancer et de reculer. Ces drones sont donc capables de s'orienter sur place - ce qui est plutôt rare dans le domaine.



FIGURE 4 : Les pompes de déplacement suivant un plan horizontal - en orange - ou vertical - en rouge

3.2.3 Le contrôle du drone

Pour contrôler ces pompes et cette hélice, une interface graphique dessinée et conçue par GRAALTECH – le fabriquant du FOLAGA – permet d’envoyer la commande à un drone. Cette commande contient un pourcentage positif ou négatif pour chaque paire de pompe – avant haut/avant bas, avant gauche/avant droit, arrière haut/arrière bas, arrière gauche/arrière droit. Au cours du stage, j’ai regardé quelles étaient les commandes envoyées par la GUI au drone – c’est à dire à quelle adresse IP et quel port – et nous avons donc pu les recopier pour contrôler l’AUV sans utiliser la GUI⁷.

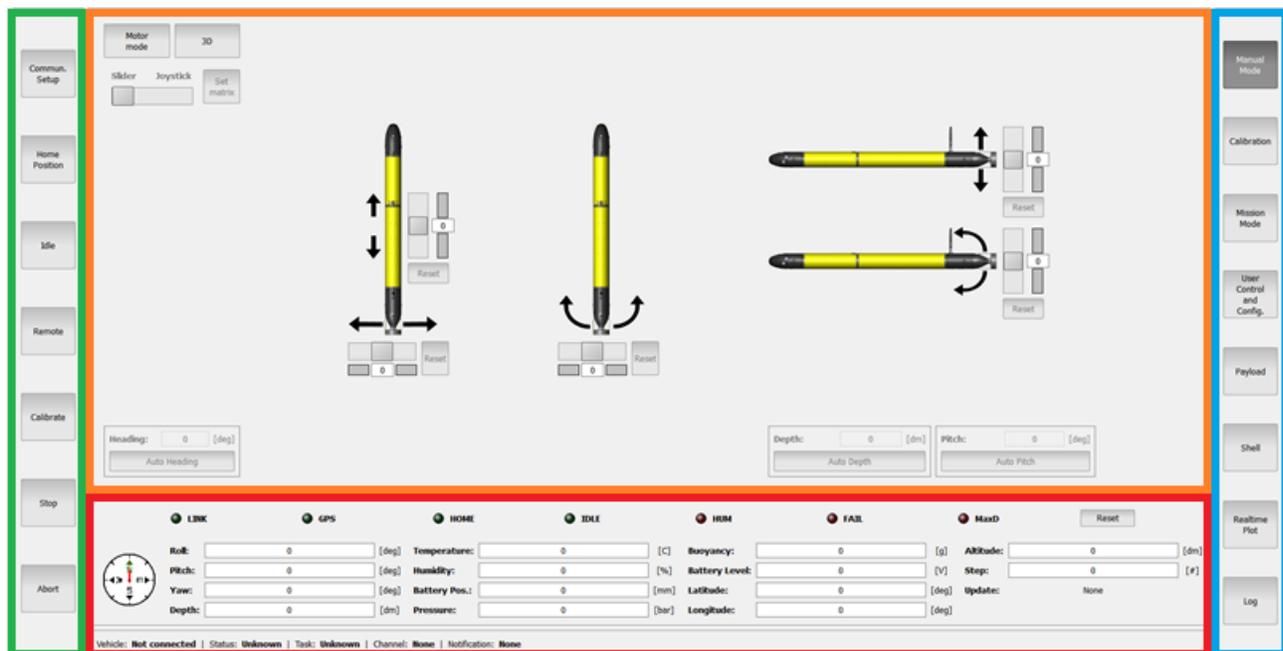


FIGURE 5 : La GUI permettant le contrôle des drones

L’interface graphique possède un certain nombre de boutons permettant de choisir dans quel mode on ordonne à l’AUV de se mettre – on y trouve par exemple le mode *calibrate* ou encore *remote*. Ces boutons se trouvent sur la partie gauche de l’interface – encadrée en vert dans la figure 5. A l’intérieur du cadre orange, nous avons accès aux réglages de l’AUV correspondant à l’onglet ouvert par les boutons présents sur la partie droite – encadrée en bleu sur la figure. On trouve parmi ces modes, celui de *calibration*, de *mode manuel* qui correspondent aux deux modes AUV en exemple. Enfin, toutes les informations de l’AUV se trouvent dans la partie inférieure – encadrée en rouge. On trouve ici des informations d’attitude comme le cap, mais aussi des informations de sécurité comme

⁷Graphical User Interface – Interface graphique

le pourcentage d'humidité, parmi beaucoup d'autres.

D'un point de vu purement technique, pour contrôler la puissance des pompes, nous utilisons une commande appelée *AXISREF* et qui empêche d'actionner deux pompes opposées en même temps. Elle prend en paramètre les pourcentages voulus. Pour les pompes qui permettent un déplacement vertical, un pourcentage positif active les pompes du dessus du drone et fait donc descendre la partie avant et/ou arrière. Un pourcentage négatif provoque l'effet inverse. Concernant les pompes horizontales, un pourcentage positif active les pompes à babord - les pompes de gauche dans le référentiel du drone. Pour la propulsion, cela est plus intuitif, un pourcentage positif fait avancer l'AUV et inversement pour une commande négative.

Il est important de noter que pour un pourcentage inférieur à environ 15%, les pompes n'arrivent pas à éjecter un jet d'eau assez puissant pour le sentir à la main. Cependant, il y a tout de même un flux d'eau entrant dans la pompe que l'on peut sentir entre 5% et 15%. Il y a une zone morte autour de 0%. De plus, la durée entre l'envoi de la commande et l'activation des pompes a été mesurée à environ 2 secondes. Ce drone possède donc une dynamique plutôt lente.

3.2.4 Les *payload*

Ces drones sont composés de sections facultatives au fonctionnement propre du drone. Elles sont appelées *payload*. Il est donc possible d'ajouter et d'enlever des *payload* des drones. Cela permet de les équiper avec des capteurs différents pour des missions également différentes. Par exemple, tous les drones sont équipés d'une tête acoustique qui permet de se localiser sous l'eau. Néanmoins, les drones peuvent fonctionner sans. Il est possible d'exécuter des missions de suivi de cap, de suivi de profondeur pendant un certain temps.



FIGURE 6 : Payload acoustique (délimitée visuellement par un changement de couleur)

KOPADIA possède une *payload* sideScan qui permet de générer des images des distances du fond marin, ce qui devrait permettre de repérer des objets. Tous les drones sont équipés d'une centrale inertielle, d'un GPS, d'un capteur de pression, d'un thermomètre, d'un capteur d'humidité et d'une tête acoustique. Ces capteurs permettent un fonctionnement qui assure une certaine sécurité pour le drone grâce à une intelligence interne au FOLAGA qui lui ordonne de remonter en cas d'urgence.

La protection de couleur grise sur la figure 6 protège une prise *subcon* étanche, pratique pour connecter un nouveau capteur à l'extérieur du drone. C'est à une prise de ce type que l'écho sondeur est connecté.

3.3 L'écho sondeur

L'écho sondeur, acheté par KOPADIA avant mon arrivée en stage, est préconisé pour une utilisation sur des ROV⁸. Il est assez compact et de prix plutôt abordable, ce qui permet à l'entreprise de l'acheter et de pouvoir ainsi s'essayer à ce pour quoi elle a été créée : ajouter des capteurs à un drone pour inspecter les fonds marins.



FIGURE 7 : Écho sondeur Imagenex 852 Ultra-miniature

3.3.1 Son fonctionnement

Le fonctionnement de cet écho sondeur est plutôt simple, il faut envoyer une certaine trame en hexadécimale comportant différents paramètres chacun représenté par un octet et l'écho sondeur renvoie une autre trame comportant les informations correspondantes à la demande. Pour connaître régulièrement l'altitude du drone, il est nécessaire d'envoyer régulièrement une trame .

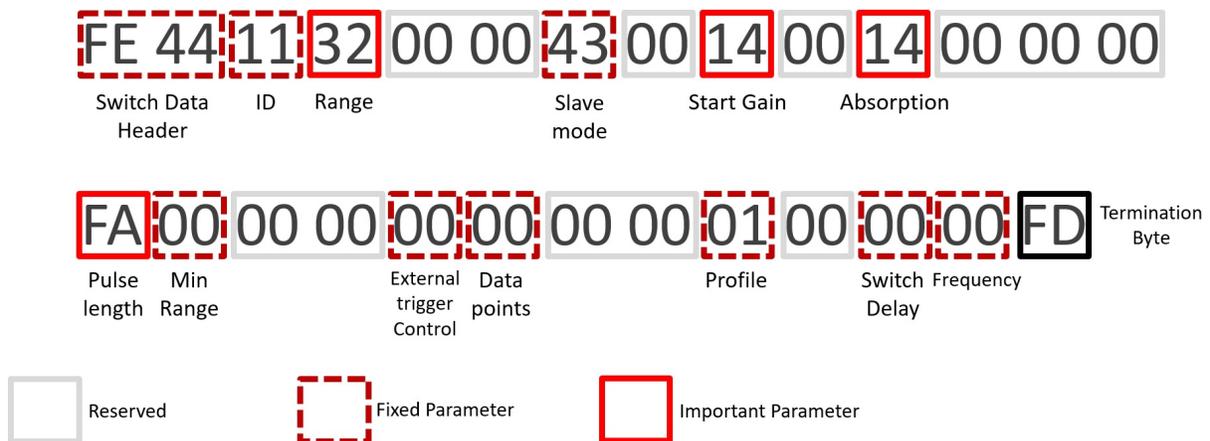


FIGURE 8 : Exemple de trame émise à l'écho sondeur

⁸ROV : Remotely Operated Vehicle - robot sous-marin contrôlé à distance

Réglages de la trame d'émission : Tout d'abord, la trame débute par deux octets qui sont 0xFE et 0x44 qui sont les bytes 0 et 1. Elle finit par 0xFD au byte 26. Il est interdit d'utiliser la valeur 0xFD pour une autre utilisation que pour signaler la fin de la trame. Entre le byte 1 et le byte 26, il y a différents réglages possibles. Voici les réglages principaux que j'ai dû modifier pour pouvoir avoir un retour d'information cohérent avec la réalité ainsi que suffisamment de valeurs.

ID - Mode (byte 2) : Le sondeur permet d'utiliser 3 ID différentes : IPX, IMX et IGX. Le premier ne renvoie qu'un seul point tandis que les deux suivants renvoient respectivement 250 et 500 valeurs de fond. Le dernier semble donc le plus intéressant. Cependant, la documentation étant beaucoup plus détaillée pour le mode IPX que pour les deux autres modes et n'ayant que peu de temps avant les premières lectures à Brest, j'ai opté pour le mode IPX. Par ailleurs, il permet une lecture plus rapide de la trame reçue puisqu'il renvoie beaucoup moins de données. Ce mode est donc plus facile à traiter pour un ordinateur intégré. Pendant la suite de mon stage, je n'ai pas pris le temps de comprendre en détails le fonctionnement logiciel des deux autres modes pour développer un suivi de fond.

Profondeur maximale (byte 3) : Dans un cadre d'utilisation courante lors de mon stage à KOPADIA, une profondeur maximale de 5m suffisait.

Gain initial (byte 8) : Comme nous travaillons, pour l'instant, presque uniquement au lac de Polytechnique avec une profondeur maximale de 2 m et un sol plutôt vaseux, il est nécessaire d'avoir un gain plutôt fort - entre 15 et 20 dB. Il faut donc mettre entre 0x0F et 0x14 à l'octet correspondant.

Longueur du signal (byte 14) : L'algorithme de détection fonctionne mieux avec un signal long. C'est pourquoi j'ai choisi de mettre ce paramètre au maximum de manière à avoir une meilleure précision plutôt qu'une grande rapidité de fonctionnement. En effet, le drone ayant plutôt une dynamique lente, il n'est pas nécessaire d'avoir des informations très fréquemment. J'ai donc mis l'octet correspondant à la valeur 0xFA qui correspond à 250ms.

Lecture de la trame reçue : Pour lire la trame, il faut dans un premier temps s'assurer qu'elle commence bien par 0x495058 – qui correspond au mode IPX – qu'elle soit de la bonne longueur – cette longueur dépend de la trame émise – et qu'elle finisse bien par 0xFC. Nous avons utilisé uniquement le mode IPX et à l'aide des bytes 8 et 9, il est possible de retrouver l'altitude envoyée par le sondeur. Dans un premier temps, nous calculons le "profil bas" et le "profil haut". L'un représente les 8 bits de poids faible et l'autre les 8 bits de poids fort. Pour le "profil bas" voici l'opération logique à effectuer :

$$(\text{byteNine} \ \&0x01) \ll 7 | (\text{byteEight} \ \&0x7F)$$

Pour le "profil haut" c'est cette opération logique qu'il faut appliquer pour retrouver le profile

$$(\text{byteNine} \ \&0x7E) \gg 1$$

On obtient ensuite l'altitude en faisant l'opération logique *ou* à ces deux "profils" en ayant au préalable décalé vers la gauche celui représentant les bits de poids fort.

3.3.2 L'intégration matérielle

Ayant une contrainte de temps majeure puisque nous avons reçu l'écho sondeur le jeudi 17 mai et que nous voulons l'utiliser à Brest le mardi 22 mai, que nous souhaitons profiter pleinement de cette occasion, nous avons donc choisi de l'intégrer à l'extérieur du drone. Gaëtan FONTAINE a conçu une protection qu'il a ensuite imprimée en 3D. Il a réalisé un *potting* pour avoir une fiche *subcon* étanche au lieu d'un port série qui ne l'est pas. Nous sommes arrivés à Brest en ayant à connecter la fiche *subcon* – à laquelle le sondeur est branché – à la carte de commande Raspberry PI 3 que nous avons placée dans la payload triton. Cette solution permet d'avoir un accès aisé à la carte en cas de maintenance à faire. Elle est positionnée à un endroit stratégique puisque les fiches *subcon* sont juste à côté, un câble USB permettant anciennement la connexion de la carte de positionnement permet maintenant l'alimentation de la Raspberry, un câble Ethernet assure aussi la communication avec la carte de commande de l'AUV. Pour finir, il y a de la place à cet endroit et la payload acoustique est presque toujours présente sur chaque drone. Tout semble nous dire que cet emplacement est le bon.

Avec Romain DUVAL nous avons eu quelques difficultés concernant la liaison avec la carte, à cause d'une différence de niveaux logiques entre la liaison série RS232 et ceux de la Raspberry. Nous n'avions pas de signal retour. Messieurs Fabrice LEBARS et Clément AUBRY nous ont aidés en nous expliquant cela et en nous prêtant un adaptateur RS232 vers USB pour avoir les mêmes niveaux logiques. Cela nous a permis de terminer plus rapidement l'intégration matérielle. Il restait à vérifier que la partie logicielle fonctionnait.

3.3.3 L'intégration logicielle

Dans un premier temps, à cause de ce planning serré et par habitude de travailler avec Python, j'ai réalisé un algorithme qui permet d'acquérir les données du sondeur. Pour cela on ouvre une liaison série avec le sondeur et on lui envoie la trame décrite précédemment. On lit la trame réponse et on ajoute au fichier initialement créé pour cet enregistrement. Cependant, il faut tout de même lancer le programme dès le démarrage de la Raspberry car il est difficile de lancer l'acquisition des données sur un zodiac en pleine mer. La communauté Raspberry étant très active, il nous a été relativement aisé de réaliser ceci avec Romain DUVAL.

3.3.4 Résultats

Lecture à Brest : Nous sommes allés à Brest le mardi 22 et le mercredi 23 mai. Nous avons fini d'intégrer le sondeur le mercredi en début d'après-midi. Nous avons pu faire un seul enregistrement de données à Brest. Ce jeu de données était cohérent avec la trame d'émission. Cependant, nous n'avons pas modifié la profondeur maximale acquise qui était de 5m. Nous n'avons donc des données que sur un très court instant d'une minute alors que nous avons une mission d'environ 20 minutes. La figure 9 présente l'enregistrement de ces données.

On observe assez facilement que le ratio de données utilisables est faible. Cependant, en recoupant ces informations à la profondeur enregistrée, ce jeu de données est compréhensible. Cette mission devait maintenir le drone à une certaine profondeur – 5m. Néanmoins, il est descendu jusqu'à 16m suivant le baromètre intégré. Ce pic de profondeur correspond avec la perte de données que l'on peut observer sur la figure 10 entre le temps 3120 et 3150. Le sondeur, avec les réglages installés n'a donc pu lire les données en deçà de 1,3m.

Malgré la déception d'avoir un jeu de données presque vierge et inutilisable – car le positionnement acoustique n'a pas fonctionné non plus – nous avons pu constater que nous avons réussi à intégrer l'écho sondeur

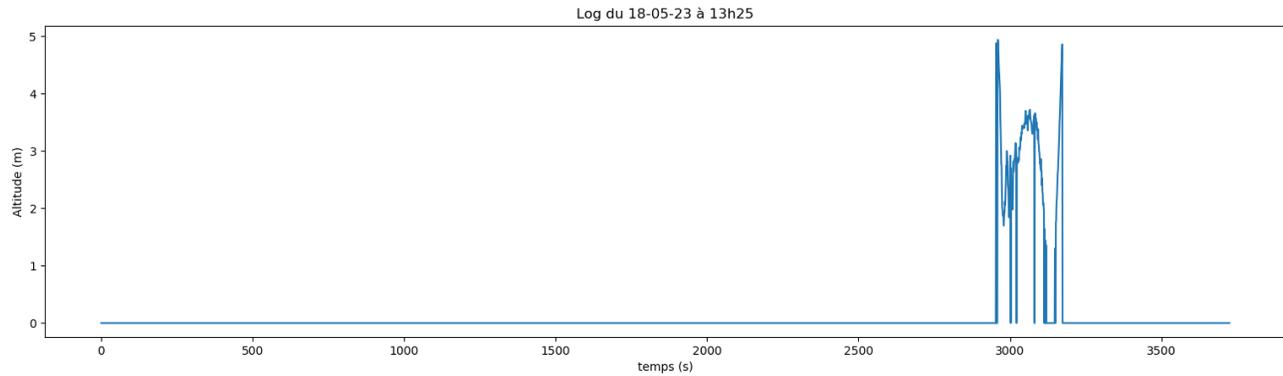


FIGURE 9 : Altitude du drone lue à Brest lors d'une mission

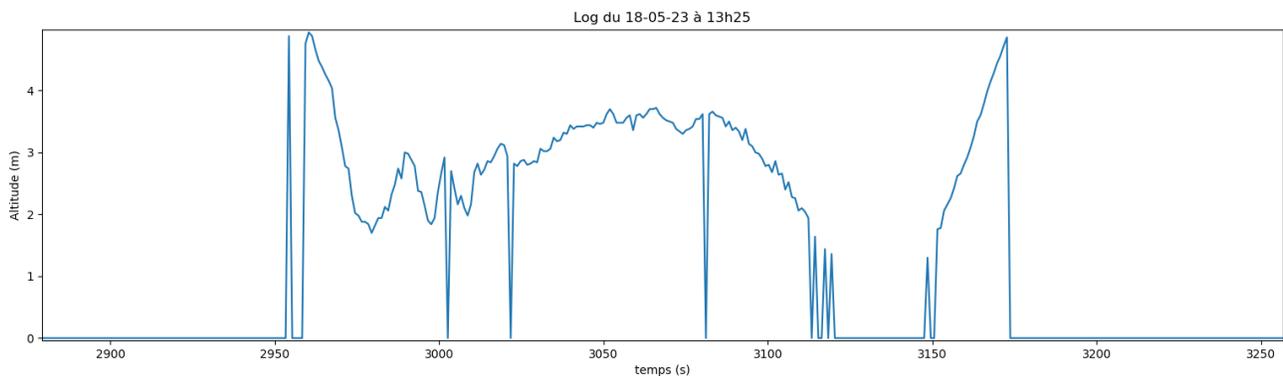


FIGURE 10 : Zoom sur l'altitude du drone lue à Brest

correctement et avoir la preuve que celui-ci fonctionnait. On ne peut pas conclure quand à la précision de celui-ci avec un si petit jeu de données et sans connaître le fond marin.

Lecture au lac de Polytechnique : Au retour de Brest, nous sommes allés au lac de Polytechnique pour tester la précision de ce capteur. Nous avons obtenu beaucoup de valeurs incohérentes – des valeurs de plus de 3m – ainsi que beaucoup de 0. Le capteur renvoie une altitude de 0 cm lorsqu'il n'arrive pas à calculer la distance. Pour régler ce problème, nous avons doublé la longueur du signal et triplé le gain initial. Nous avons donc une longueur de 250ms et un gain de 20dB. Nous pouvons donc nous focaliser sur la seconde tâche principale du stage : le suivi de fond.

3.4 Mission Oriented Operating Suite - MOOS

Pour réaliser le suivi de fond, nous devons enregistrer les différentes altitudes, les analyser pour obtenir une commande cohérente et l'envoyer au drone pour qu'il l'exécute. Nous avons donc besoin d'une base de données et d'un certain nombre de petites applications qui viennent récupérer des entrées de la base de données et écrire d'autres entrées. Nous avons donc besoin d'un *middleware*. Nous avons utilisé MOOS ou plutôt MOOS-IvP⁹. MOOS a l'avantage d'avoir une documentation très complète et une utilisation plutôt facile. MOOS-IvP est une extension de MOOS et présente des outils de communication entre les drones.

⁹IvP signifie InterVal Programming : Programmation par intervalle

3.4.1 Le fonctionnement

MOOS – et par extension MOOS-IvP – est basé sur une architecture *publish-subscribe*. Cela signifie que chaque application s’enregistre à une liste de diffusion d’un certain nombre de variables de la base de données et reçoit les mises à jour de celles-ci. De plus, chaque application publie un certain nombre de variables dans la base de données et les met à jour si elles sont déjà existantes.[6]

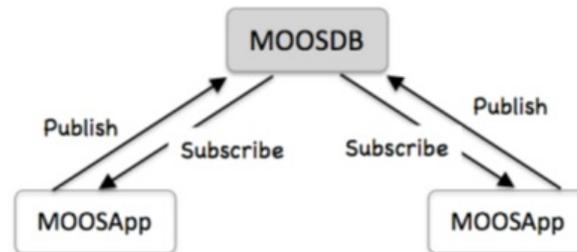


FIGURE 11 : Fonctionnement de MOOS

MOOS est implémenté en C++. Il est possible de l’utiliser avec Python mais comme Oriane et Romain n’ont presque jamais fait de Python, nous avons choisi le C++.

Ce middleware exécute des applications qui communiquent entre elles via une base de données. Chaque application MOOS est décomposée en plusieurs fichiers mais un seul peut être modifié pour développer les applications. Voici une description des méthodes nécessaires au fonctionnement de l’application MOOS implémentée.

Le constructeur : Initialisation des variables et initialisation des connexions si nécessaire.

Le destructeur : Libération de l’espace utilisé.

Au nouveau mail : Permet de récupérer les valeurs mises à jour dans la base de données de MOOS.

A la connexion au serveur : Lance la souscription aux variables nécessaires.

Itération : Cette méthode est exécutée un nombre de fois défini dans les fichiers missions. C’est ici que l’on implémente les algorithmes de navigation.

Au démarrage : Récupère les paramètres nécessaires au fonctionnement de l’application.

Souscription aux variables : C’est ici que l’application s’enregistre aux variables souhaitées.

Pour tester une application développée ou pour exécuter une mission, il est nécessaire de créer un fichier de mission d’extension *.moos*. Voici donc le format d’un de ces fichiers.

Dans chaque fichier mission il doit y avoir l’adresse et le port sur lequel la mission doit être lancée. Ensuite, il faut configurer les applications exécutées dans MOOS avec *ANTLER*. On peut choisir le temps entre chaque lancement d’application ainsi que les applications qui doivent être exécutées. L’application MOOSDB doit être lancée pour permettre aux autres applications de se partager les informations. Pour les applications qui sont exécutées dans la mission, il faut aussi les configurer. Pour la plupart, définir la fréquence d’exécution – le *AppTick* – et la fréquence de communication à la MOOSDB – le *CommsTick*.

```
1 ServerHost = localhost
2 ServerPort = 9000
3 Community = alpha
4
5 ProcessConfig = ANTLER
6 {
7     MSBetweenLaunches = 200
8
9     Run = MOOSDB @ NewConsole = false
10    Run = pSonderLecture @ NewConsole = false
11    Run = pAcquireFolaga @ NewConsole = false
12
13 }
14
15 ProcessConfig = pAcquireFolaga
16 {
17     AppTick = 4
18     CommsTick = 4
19 }
20
21 ProcessConfig = pSonderLecture
22 {
23     AppTick = 4
24     CommsTick = 4
25
26 }
27
```

FIGURE 12 : Exemple de fichier mission du test de la lecture sondeur

3.4.2 Architecture choisie

Nous devons choisir une architecture pour notre code. Nous avons décidé d'avoir des applications, qui s'occupent uniquement de l'interface avec les capteurs et avec les drones, qui calculent la prochaine commande en fonction des données des capteurs ainsi que d'une application qui récupère les variables de commande et crée la chaîne de caractères prête à être envoyée. La figure 13 montre le cheminement du calcul au travers MOOS-IvP. Les applications en *jaune* sont des applications interfaces, celles en *vert* traitent les données et celles en *bleu* calculent la commande.

Toutes les applications présentes sur la figure 13 sont donc reliées à la base de données de MOOS également appelée MOOSDB. Les applications qui permettent de recueillir des données sont en *jaune* sur la figure 13, en *vert* sont les applications qui filtrent, mettent en forme les données utilisées ou renvoyées par les programmes de pilotage, de commande en *bleu*.

pSonderAltitude : Ce programme lit l'altitude du drone – c'est à dire la distance entre le fond et le drone. C'est celui qui envoie la trame décrite plus haut et qui traite la trame réponse pour insérer l'altitude brut de l'AUV dans la MOOSDB.

pFilterAltitude : Cette application applique un filtre à l'altitude lue par le sondeur. L'altitude renvoyée est moins bruitée et plus fiable.

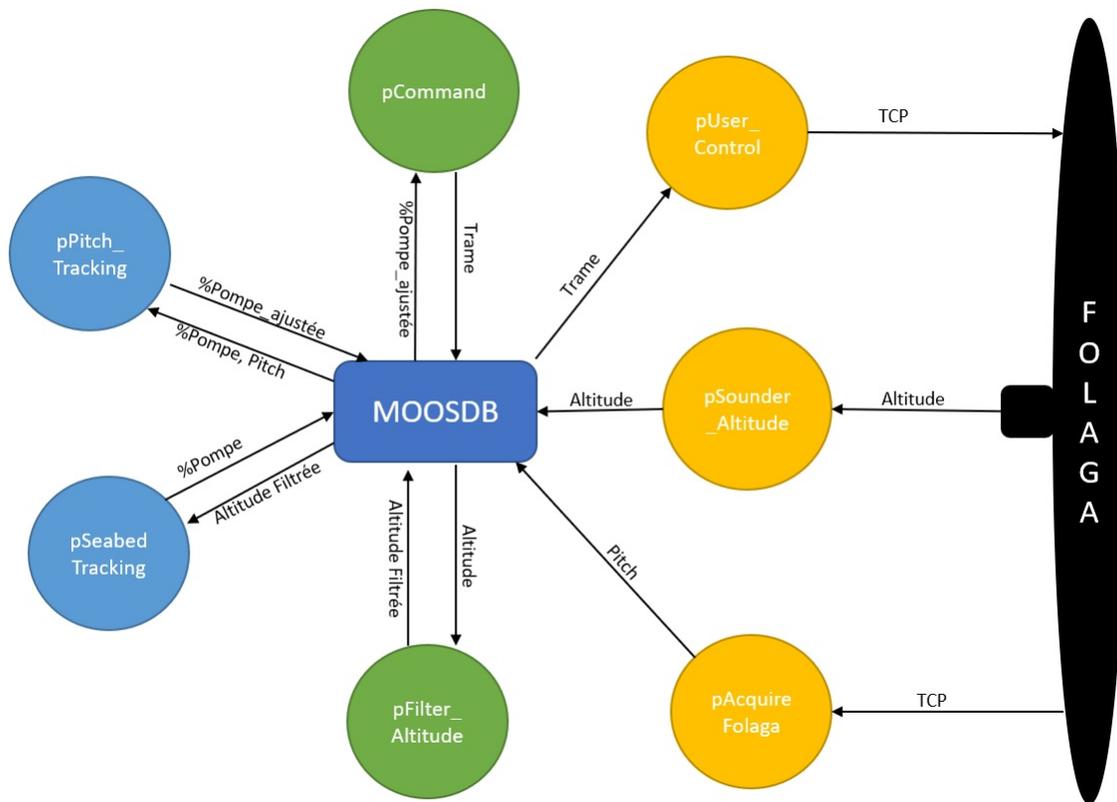


FIGURE 13 : Architecture de développement de MOOS

pSeabedTracking : Ce nœud est celui qui permet à l'AUV de se réguler en altitude et donc d'exécuter un suivi du fond marin. Pour cela nous avons implémenté un PID¹⁰. N'ayant que peu de connaissance sur le drone, nous avons décidé de commencer par faire un PID pour avoir un suivi de fond fonctionnel. C'est-à-dire qu'il n'est pas parfait mais réagit suffisamment pour répondre aux besoins du moment de l'entreprise.

pAcquireFolaga : Cette brique technologique, que Romain DUVAL a majoritairement développée, permet d'acquérir l'assiette de l'AUV via une connexion *tcp*. Cette mesure n'est pas très juste mais semble relativement fiable, quand le drone est en surface sur l'eau, il a une assiette d'environ -2 degrés. Nous avons tenté de calibrer l'IMU¹¹ mais le résultat en assiette n'a pas été amélioré alors que le cap est un peu plus juste qu'avant.

pPitchTracking : Ce module est particulièrement important car il régule l'assiette de l'AUV. Ceci influence directement sur le suivi de fond puisque lorsque le drone avance à sa vitesse maximale, une assiette non nulle le fait descendre ou monter et les pompes verticales agissent alors moins sur la correction d'altitude. Il récupère donc la commande des pompes du suivi de fond et corrige cette dernière en fonction de l'assiette lue par la centrale inertielle.

¹⁰Proportionnel Intégral Dérivée : commande proportionnelle : à l'erreur, à la dérivée de l'erreur ainsi qu'à son intégrale

¹¹Une IMU est une centrale inertielle et permet d'obtenir des informations comme l'assiette, la gîte et le cap

pCommand : Ce nœud qu'Orianne ADAMM a développé, récupère les différentes commandes des précédentes applications et les combine pour créer une chaîne de caractères qu'il restera à envoyer à l'AUV. C'est l'application qui gère toutes les commandes à envoyer au FOLAGA. Elle souscrit à toutes les variables de commande de MOOSDB pour pouvoir communiquer ce qui est nécessaire au drone.

pUserControl : Cette application développée aussi par Orianne ADAMM envoie à l'AUV un paquet suivant le protocole de connexion *UDP* pour le mettre en mode contrôle puis envoie grâce à une connexion suivant le protocole *telnet* la chaîne de caractère générée par l'application *pCommand*.

3.5 Régulation d'assiette

3.5.1 Motivations et cahier des charges

La gestion de l'assiette n'étant assurée qu'à minima par la mécanique du drone – de part sa forme, le drone a tendance à se remettre à l'horizontal s'il n'avance pas, mais il ne s'équilibre pas assez vite pour pouvoir corriger l'assiette que lui procurera une avancée – de part sa dissymétrie de section de face.

La table 1 présente la distance verticale parcourue en une seconde par l'AUV à sa vitesse maximale – 1m/s.

Assiette (degrés)	0	1	2	3	4	5	6	7	8	9	10
Vitesse verticale (cm/s)	0	1.7	3.5	5.2	7.0	8.7	10	12	14	15.5	17

TABLE 1 : Influence de l'assiette sur l'altitude du drone

En plus d'avoir une lecture de l'altitude, via l'écho sondeur, qui n'est pas exactement sous l'AUV, celui-ci remonte ou redescend en fonction de son assiette. Nous voulons dé-corréler au maximum la commande de l'avancée de la commande verticale. Il est donc nécessaire d'avoir une régulation en assiette qui empêche que le drone prenne une assiette importante.

Au vu des capacités du FOLAGA, nous avons donc choisi arbitrairement que l'assiette devienne vraiment importante lorsque la distance verticale parcourue en une seconde était de plus de 10cm. Cela correspond donc à un angle maximum de 6 degrés.

Le cahier des charges est donc simple, le drone ne doit pas sortir de la zone +/- 6° et doit rentrer dans cette zone de tolérance le plus vite possible si jamais il n'y est pas déjà.

Cependant, on peut supposer qu'en temps d'utilisation normal, le drone est envoyé depuis la surface en mission. Or en surface il est à une assiette que l'on peut considérer comme nulle – mis à part les vagues qui pourraient lui faire prendre de l'assiette. Il suffit donc d'empêcher le drone de s'incliner. Ayant choisi de réaliser un PID pour les mêmes raisons que le suivi de fond, il conviendra d'avoir une dérivée forte pour que tout changement d'assiette entraîne une correction suffisante pour maintenir le drone dans la zone tolérée.

3.5.2 Réglage du PID

Au vu de la section précédente, nous avons convenu qu'un PID avec une dérivée forte permettrait de répondre à notre besoin. Cependant, le drone a un temps de réaction d'environ 2 secondes, donc au moment où la commande est perçue sur l'assiette, il s'est écoulé 2 secondes. Il est donc nécessaire d'avoir des coefficients qui viennent aider

la dérivée à maintenir le drone dans une position horizontale. La table 2 montre les coefficients qui m'ont semblé intéressants pour avoir une réponse cohérente avec le besoin.

Proportionnelle	3
Dérivée	31
Intégrale	0.09
Maximum Intégrale	400

TABLE 2 : Coefficients finaux de réglage du PID de la régulation d'assiette

Pour trouver ces coefficients nous avons dans un premier temps mis le drone dans le bassin d'essai devant l'atelier, plongé celui-ci en activant les pompes verticales à 50% tout en étant maintenu à une certaine distance de la surface grâce à une corde. Pour tester l'efficacité du réglage, nous le poussons d'un des deux côtés et nous observons le temps de réponse, l'oscillation, l'erreur statique, le temps pour qu'il ne sorte plus des 5° de flexibilité. A la suite de ce réglage, nous avons légèrement modifié les coefficients lors des tests de suivi de fond puisque le but de cet algorithme est d'empêcher le FOLAGA de se pencher lors d'un déplacement.

La figure 14 présente la courbe de l'assiette lors d'une mission de suivi de fond. Il est important de noter que l'IMU étant mal calibrée, le drone est à l'horizontal quand l'assiette vaut -2°.

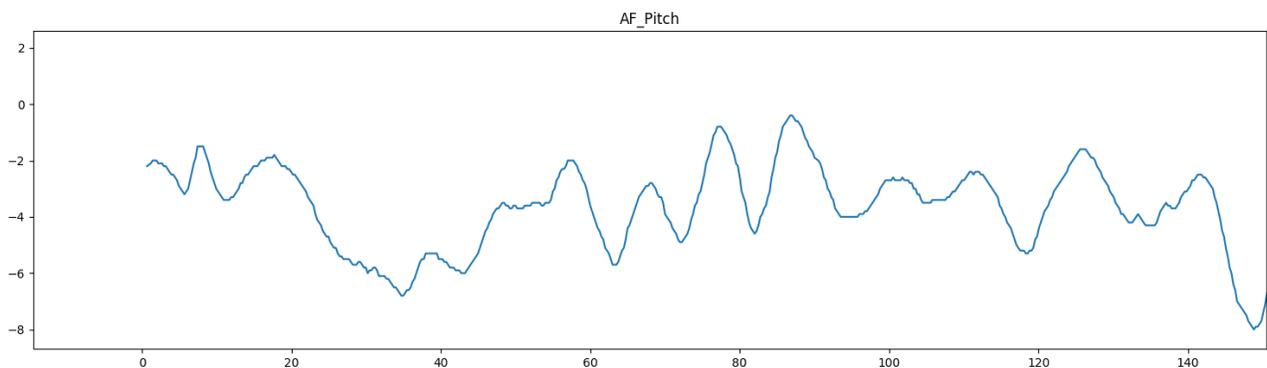


FIGURE 14 : Graphique de la régulation en assiette

On peut noter que la moyenne de ce jeu de données est inférieure à la commande ce qui fait que le drone a tendance à descendre dans les profondeurs lors de son avancée. Nous avons jugé que cela était tout de même acceptable et que nous pouvions régler le suivi de fond malgré tout.

3.6 Suivi de fond

3.6.1 Une première simulation

J'ai réalisé deux simulations de suivi de fond pendant mon stage. J'ai développé la première pendant la première semaine du stage. Elle a permis de montrer ce qui était possible théoriquement sans les contraintes des FOLAGA et des limites des capteurs car les premiers, je ne les connaissais pas puisqu'ils étaient en Italie chez leur constructeur GRAALTECH pour une maintenance et le second, nous ne l'avions pas encore reçu. Cependant ayant une

description de leur mode de fonctionnement, j'ai pu trouver leur équation d'état en négligeant les frottements :

$$\begin{cases} \dot{x}_1 = x_3 \\ \dot{x}_2 = x_4 \\ \dot{x}_3 = u_1 \\ \dot{x}_4 = u_2 \end{cases}$$

En appliquant un contrôle avec une proportionnelle et une dérivée, ainsi qu'une zone d'inactivité – en bleu sur la figure 15 – autour de la commande – en vert – et le drone - en noir.

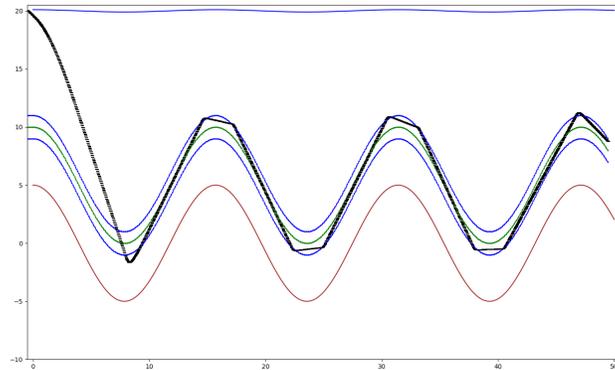


FIGURE 15 : Simulation du suivi de fond par l'AUV

3.6.2 Une deuxième simulation

Après cette première simulation, nous avons besoin de connaître davantage la dynamique des FOLAGA. La figure 16 présente cette simulation du suivi de fond, plus réaliste avec une dynamique plus réelle du drone.

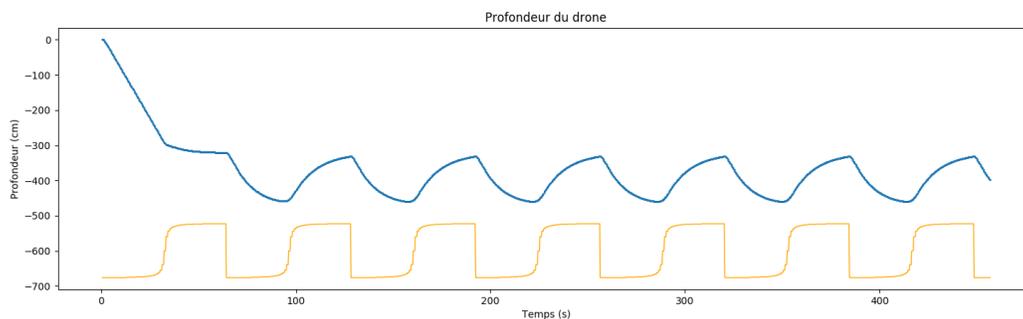


FIGURE 16 : Suivi de fond simulé avec une dynamique corrigée

Voici donc cette étude.

Nous avons donc fait une petite étude pour obtenir les forces de frottements appliquées à l'AUV pour obtenir une simulation assez proche de la réalité. Dans un premier temps, nous avons regardé les forces appliquées à l'AUV. Le schéma de ce bilan des forces est présenté en figure 17.

On étudie donc le FOLAGA qui est neutre dans l'eau – on ne considérera donc ni la poussée d'Archimède ni le poids de l'AUV. On suppose dans cette étude que la force de trainée est uniquement proportionnelle au carré

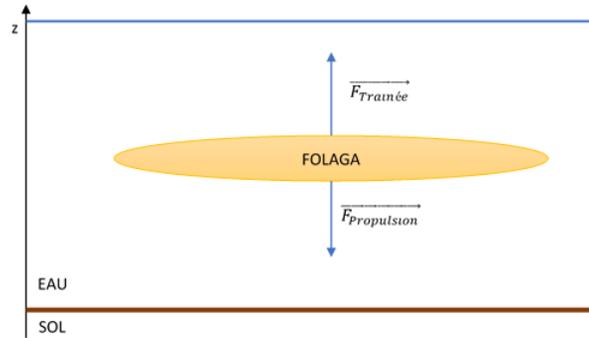


FIGURE 17 : Bilan des forces appliquées au drone

de la vitesse. On applique le principe fondamental de la dynamique au drone, ce qui nous permet d'écrire :

$$m \vec{a} = \vec{F}_{\text{Trainée}} + \vec{F}_{\text{Propulsion}}$$

En projetant suivant l'axe Oz, on obtient $\ddot{z} = A\dot{z}^2 + B(p)$ où $B(p)$ est une fonction qui dépend du pourcentage de propulsion des pompes verticales et vaut $B(p) = -\frac{\beta(p)}{m}$ et $A = \frac{\alpha}{m}$

Comme cette équation différentielle n'est pas linéaire, nous allons passer par une approche numérique à l'aide de la méthode d'Euler et ainsi obtenir une solution approchée. Cependant, la solution de cette équation différentielle admet un point fixe quand $\ddot{z} = 0$ et vaut :

$$v_{\text{limite}} = \sqrt{\frac{B(p)}{-A}}$$

On obtient donc une solution de la forme suivante :

$$v_{i+1} = v_i + (Av_i^2 + B)dt$$

On en déduit la position verticale à l'instant $i+1$.

$$z_{i+1} = z_i + v_{i+1}dt$$

Pour obtenir le coefficient de frottement α – dans le terme A – Gaëtan FONTAINE a simulé sous Solid-Works un écoulement vertical à 0.2 m/s – qu'on présume atteignable par l'AUV. Nous avons alors trouvé une force de frottement de 7N. Nous en avons donc déduit que $\alpha = 175 \text{ kg/m}$.

Pour mesurer la masse du FOLAGA nous avons utilisé une balance à main – ce n'est pas très précis mais nous avons jugé cela suffisant. Nous avons trouvé que $m = 36 \text{ kg}$.

Pour mesurer le coefficient de propulsion nous avons calibré le drone de manière à ce qu'il soit neutre dans l'eau puis nous avons pesé le drone avec la même balance à main que pour le peser dans l'air. Nous avons ensuite activé les pompes verticales de 0 à 100% par pas de 10%. Voici le tableau des résultats :

En effectuant une régression affine, on obtient que $\beta(p) = (0.02p - 0.4)g$ ou $g = 9.8 \text{ N}$.

Pourcentage	0	10	20	30	40	50	60	70	80	90	100
Masse lue	0	0	0.05	0.25	0.45	0.6	0.75	0.9	1.2	1.5	1.7

TABLE 3 : Masse lue sur le pèse main en fonction de la propulsion verticale

On peut vérifier la cohérence des résultats en observant la vitesse verticale limite lorsque les pompes sont à 100% . On obtient alors :

$$v_{limite} = \sqrt{\frac{B}{-A}} = \sqrt{\frac{-0.44}{-4.86}} = 0.29\text{m/s}$$

A la vue du comportement du FOLAGA cela nous est alors apparu cohérent. Nous avons donc décidé de valider cette simulation.

3.6.3 En réel

Ayant relativement beaucoup de points égaux à 0 ou des valeurs trop grandes par rapport à l'environnement, quand on reçoit un zéro ou quand la différence avec la valeur précédente est supérieure à 70cm, la valeur suivante est la moyenne des 2 précédentes. Ayant une lecture tous les dixièmes de seconde, avec une vitesse maximale d'environ 1m/s le drone aura avancé de 10cm. Ayant pour but de suivre une pipeline posée au sol, un dénivelé égal à 700% n'a qu'une toute petite chance d'arriver. De plus, étant un premier filtre, cette application sera amenée à être modifiée dans le futur pour encore plus être précise. La figure 18 présente donc sur un même graphique, les données brutes et filtrées de l'altitude du drone pendant qu'il avance avec la propulsion à 100%.

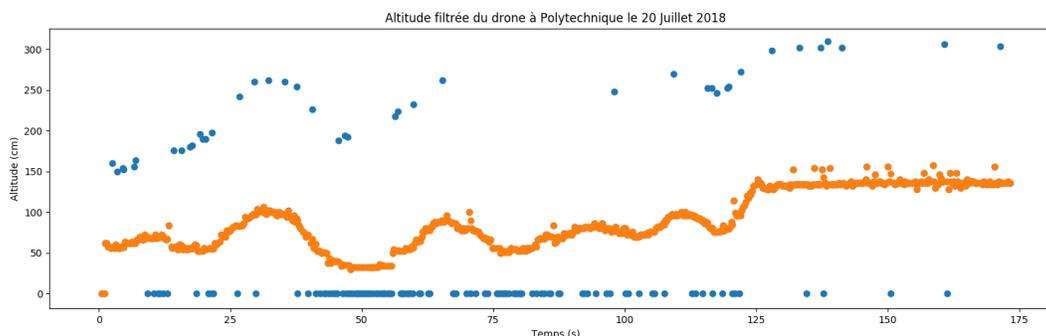


FIGURE 18 : Comparaison de l'altitude lue par le suivi de fond avec et sans filtre

On devine donc qu'en travaillant avec les données filtrées, le suivi de fond sera de meilleure qualité. Le filtre pourrait être amélioré en estimant qu'au bout d'un certain nombre de valeurs considérées comme « fausses », il réintègre tout de même les prochaines valeurs. Cependant, nous n'avons pas eu besoin d'une amélioration dans ce sens puisque lors de nos essais, nous avons toujours eu une donnée juste suffisamment fréquemment.

La figure 19 affiche un graphique de régulation de l'altitude du drone avec un PID mal réglé puis la figure 20 présente un suivi de fond avec le PID réglé.

En comparant ces deux suivis de fond, nous observons que le premier a tendance à osciller plus que le deuxième autour d'une valeur inférieure à la consigne de 100cm tandis que le suivi de fond réglé n'oscille que très peu, même s'il reste tout de même légèrement en dessous de la consigne identique.

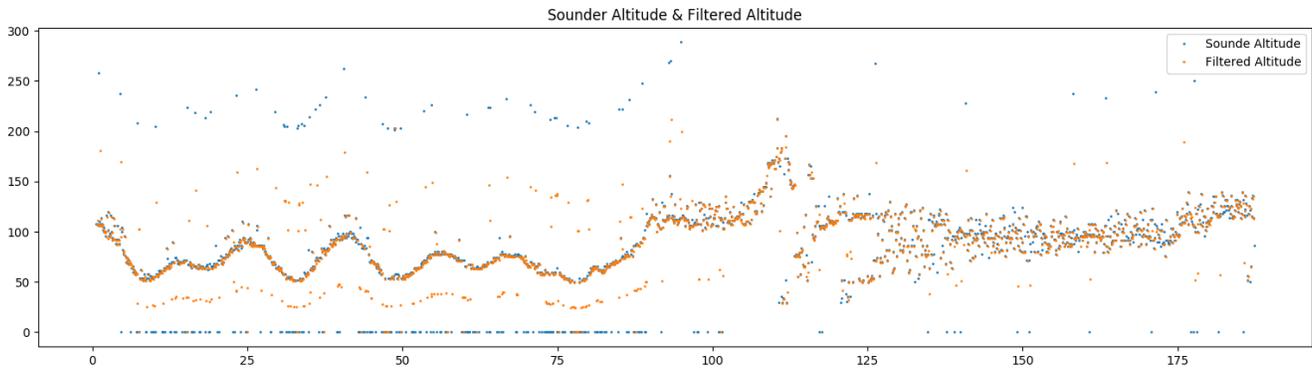


FIGURE 19 : Suivi de fond régulé par un PID – Mission de 90 secondes

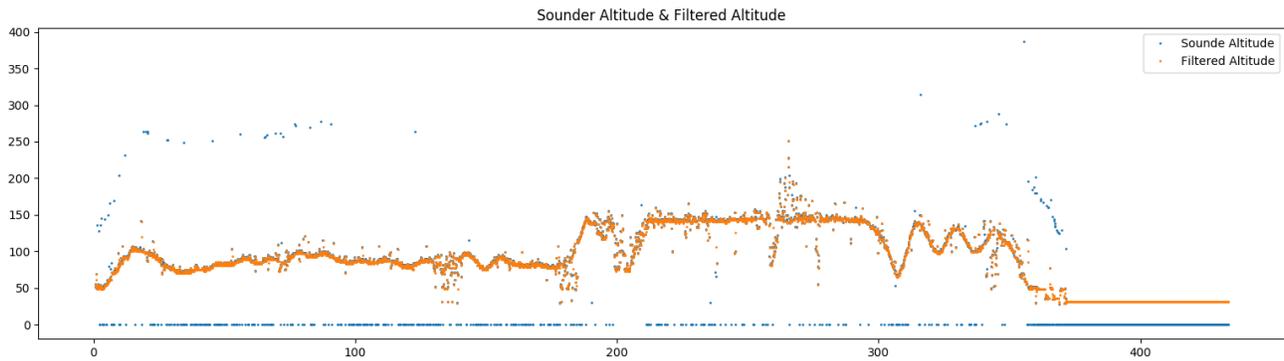


FIGURE 20 : Suivi de fond régulé par un PID – Mission de 180 secondes

Implémenter un suivi de fond qui répond à un certain cahier des charges obligeant l'AUV à rester à +/- 30cm de l'objectif n'est pas évident et est complètement lié à la bonne lecture de l'altitude. La dynamique du drone et son temps de réaction n'aidant pas à une réponse rapide et précise, la commande doit être précise. La table 4 présente les coefficients qui m'ont semblé répondre le mieux à cette demande et qui correspond à l'heure actuelle à ceux gardés par l'entreprise.

Proportionnelle	2
Dérivée	2
Intégrale	0.001
Maximum Intégrale	6000

TABLE 4 : Coefficients finaux de réglage du PID du suivi de fond

4 Analyse économique

KOPADIA étant une jeune entreprise en plein développement, ses moyens économiques sont nettement inférieurs à ceux d'un grand groupe. Le modèle économique de l'entreprise se doit d'être rigoureux pour rester dynamique.

Son fonctionnement est assez simple, il existe deux secteurs chez KOPADIA, l'ingénierie et l'inspection sous-marine. Les deux se soutiennent mutuellement d'un point de vue financier. C'est à dire que si un des deux domaines ne génère pas assez de revenus, c'est l'autre qui doit être assez fort pour financer les besoins des deux secteurs. Aujourd'hui, l'inspection sous-marine est en plein développement, c'est donc la partie ingénierie qui génère tous les revenus et l'entreprise redistribue en fonction des besoins des deux parties.

KOPADIA possède des licences exclusives sur sur 6 brevets de CGG¹² sur les modes opérationnels de meute de drone. Le but pour KOPADIA est d'utiliser plusieurs AUV en coopération pour inspecter à des coûts plus faibles que ceux du marché actuel et de manière plus sûre qu'aujourd'hui.

Le constat des fondateurs est que les systèmes actuels coûtent cher et ne sont pas robustes, c'est à dire qu'il y a une part d'incertitude importante quant à la bonne réalisation de la mission. Ces marchés qu'ils souhaitent démocratiser et fiabiliser sont ceux de la maintenance industrielle off-shore - celui-ci peut-être pour le secteur pétrolier comme pour celui des énergies maritimes renouvelables – mais aussi environnemental.

Pour le secteur éolien, par exemple, la maintenance des câbles électriques doit être à moindre coût car les éoliennes ne sont pas aussi rentables que le pétrole et doivent donc minimiser les coûts d'entretien pour rester compétitives.

Un ingénieur qui développe des moyens d'inspection sous-marine chez KOPADIA doit donc penser dans un premier temps à la simplicité d'utilisation du produit développé, à la fiabilité du système pour que la mission soit toujours réussie et enfin à ce que les solutions implémentées soient efficaces pour pouvoir faire des économies.

L'idée des deux fondateurs est de multiplier le nombre de drones nécessaires mais diminuer le nombre de tâches différentes qu'un seul drone effectue. L'intelligence du système doit donc être partitionnée entre les AUV pour augmenter la robustesse de la mission mais aussi faire des économies.

¹²CGG est une entreprise dans les géosciences, elle vend des produits et des services dans ces domaines pour des entreprises du secteur pétrolier et gazier.

5 Projet professionnel

J'ai effectué mon stage de première année dans une entreprise possédée par un très grand groupe. Dans le rapport de stage de première année, je concluais en expliquant qu'à la suite de ce stage, je souhaitais plutôt m'orienter vers des structures petites et légères. J'ai donc effectué mon stage dans une entreprise de ce type.

La facilité de communication, la répartition des tâches, les prises de responsabilité, l'autonomie, la confiance, les relations avec les différents membres du personnel, l'ambiance générale de l'incubateur ; tout concourt à me faire apprécier la manière dont fonctionne cette entreprise et je considère que c'est en effet le type de structure qui me correspond.

Lors de ce stage, j'ai pu apprendre – d'un point de vu technique – un nouveau langage de programmation, le C++, à utiliser un *middleware*, MOOS, à structurer un projet qui n'est que le commencement d'un développement conséquent.

D'un point de vue un peu plus général, j'ai aussi appris à préparer des missions de drone – que ce soit une anticipation des risques, ou bien une préparation logistique – à travailler dans des situations de mission – sur un zodiac par exemple, à anticiper les risques liés à l'activité d'une entreprise – avec la participation à la mise en place du document unique d'évaluation des risques – à développer des compétences de leadership lors de missions. Mon apprentissage pendant ce stage n'a donc pas cessé un seul jour de ce stage.

De plus, le domaine maritime – un milieu plus contraignant que beaucoup d'autres, ce qui apporte du challenge – est un domaine que j'aime. C'est donc pour toutes ces raisons que j'ai souhaité continuer dans cette entreprise en effectuant un contrat de professionnalisation chez KOPADIA. Et c'est pourquoi, je pense donc, à la suite de ce contrat, continuer chez KOPADIA si possible ou au moins dans une structure identique et toujours dans le milieu maritime.

6 Conclusion

Le but de ce stage était de réaliser un algorithme de navigation. Il fallait donc définir ce qui était le plus important et ce fut le suivi de fond. Cependant, KOPADIA ne possédait aucune structure de navigation déjà en place lors de mon arrivée. C'est pour cela que ce stage m'a beaucoup apporté, bien plus que si je n'avais développé qu'un suivi de fond. J'ai appris à régler un correcteur proportionnel intégral et dérivé, à travailler dans un petit collectif, à effectuer des essais en dehors des zones classiques de travail, à étudier les risques liés au travail, à utiliser un middleware tel que MOOS, ... La liste est longue. Ce qui m'a le plus marqué, c'est la recherche de la Cordelière à Brest. Il fallait allier efficacité, pression, sécurité, fatigue pour arriver à un résultat. Certes, ce résultat n'a pas été aussi bon que souhaité mais il a été positif tout de même. Ces essais ont aussi apporté de la cohésion à l'équipe et m'ont permis de m'intégrer plus rapidement au sein du groupe.

La vie d'une start-up comme KOPADIA n'est pas simple. Il faut se faire sa place. N'ayant pas des moyens aussi importants que des grands groupes, la priorité des fournisseurs n'est pas orientée vers ce petit budget. Il arrive régulièrement qu'il faille contacter le constructeur des drones car la documentation fournie n'est pas suffisamment développée. Cependant, la communication peut être assez compliquée, ce qui ralentit les différents projets. Par conséquent, j'ai appris à prendre contact par des voies de communications différentes. Néanmoins, comme GRAALTECH a peur que KOPADIA vole son savoir, cette entreprise souhaite fournir uniquement le minimum d'information. C'est pourquoi, à la fin de mon stage, le drone ne peut être contrôlé que pendant 10 minutes. A la suite de ce temps, il se met en sécurité et remonte.

Malgré tout, à la fin du stage, l'algorithme de suivi de fond permet de maintenir le drone sous-marin à une altitude avec une précision de +/- 30cm, ce qui est suffisant pour répondre à la demande initiale qui était de +/- 1m. Il a fallu passer par des étapes intermédiaires comme la régulation de l'assiette ainsi que mettre en place un outil permettant de partager simplement des variables, MOOS. A présent, il est possible de développer des algorithmes plus globaux.

Ce stage a une suite puisque je suis actuellement en contrat de professionnalisation chez KOPADIA. Tout reste à construire : par exemple, un mode de navigation qui intègre différents briques technologiques comme le suivi de fond et qui permettra par exemple d'explorer une zone définie au préalable.

COMPORTEMENT AU TRAVAIL / BEHAVIOUR TOWARDS WORK

Le comportement du stagiaire était-il conforme à vos attentes (Ponctuel, ordonné, respectueux, soucieux de participer et d'acquérir de nouvelles connaissances) ?

Did the trainee live up to expectations? (Punctual, methodical, responsive to management instructions, attentive to quality, concerned with acquiring new skills)?

A B C D E F

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here* _____

INITIATIVE – AUTONOMIE / INITIATIVE – AUTONOMY

Le stagiaire s'est-il rapidement adapté à de nouvelles situations ?

(Proposition de solutions aux problèmes rencontrés, autonomie dans le travail, etc.)

A B C D E F

Did the trainee adapt well to new situations?

(eg. suggested solutions to problems encountered, demonstrated autonomy in his/her job, etc.)

A B C D E F

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here* Fais bonne adaptation dans un

environnement changeant

CULTUREL – COMMUNICATION / CULTURAL – COMMUNICATION

Le stagiaire était-il ouvert, d'une manière générale, à la communication ?

Was the trainee open to listening and expressing himself/herself?

A B C D E F

Souhaitez-vous nous faire part d'observations ou suggestions ? / *If you wish to comment or make a suggestion, please do so here* _____

OPINION GLOBALE / OVERALL ASSESSMENT

❖ La valeur technique du stagiaire était :

Evaluate the technical skills of the trainee:

A B C D E F

III - PARTENARIAT FUTUR / FUTURE PARTNERSHIP

❖ Etes-vous prêt à accueillir un autre stagiaire l'an prochain ?

Would you be willing to host another trainee next year? oui/yes non/no

Fait à Orsay, le 26.07.2012

In _____, on _____

Signature Entreprise _____ Signature stagiaire

Company stamp _____ Trainee's signature



Merci pour votre coopération
We thank you very much for your cooperation

Références

- [1] CapTronic & Alciom. Introduction aux techniques de régulation. 2016.
- [2] Argos-system. Améliorer la collecte de données et les prévisions océaniques. <http://www.argos-system.org/fr/applications-argos/oceanographie/>, 2018. [En ligne; dernier accès le 28-Juillet-2018].
- [3] Ocean Infinity. Search for malaysia airlines flight mh370. <https://oceaninfinity.com/mh370/>, 2018. [En ligne; dernier accès le 28-Juillet-2018].
- [4] MIT. Moos-ivp. <http://oceanai.mit.edu/ivpman/pmwiki/pmwiki.php?n=Lab.HomePage2680>, 2017. [En ligne; dernier accès le 21-Juillet-2018].
- [5] Petrus. Petrus & eca group : Significant partnership and positioning in subsea robotics services (auvs and rovs) for the oil & gas market. <http://www.petrustech.com/>, 2018. [En ligne; dernier accès le 21-Juillet-2018].
- [6] Simon Rohou. Présentation du middleware moos-ivp. In *Middleware, Autonomie, Prise de Décisions*, 2016.